Pixel Readout

The pixel readout sequence is fully documented in the RHVD homepage. There you can find very interesting information about the pixel readout.

http://www-rhvd.fnal.gov/

Trigger Logic

The trigger for the pixel boards has basically three different trigger sources: TFIB_trigger, Chip_or and RS485 interface.

The TFIB_trigger (low true) when received by the outer board will initiate a readout, even if the chip has no data. In this case an empty event will be readout.

The Chip_or trigger was implemented to allow us to use a radioactive source or a pulse generator as the source of the trigger. When the outer board recognizes a hit (Chip_or = 0), it will generate a trigger to the STAR, and wait 100us for the TFIB_trigger to come back from the TFIB. After this time the outer board will one more time send a request to the STAR, and wait another 100us. If no response is received, after 200us the pixel will be reset, and Chip_or will go to 1.

A third way to generate the trigger was used during the characterization of FPIX0, where it was desired that all the pulses injected in the chip could generate a readout request to STAR. In this way the threshold dispersion could be analyzed. It is done using a signal from the RS485, which is buffered in the Altera part. This signal is then sent to the trigger input in the back of the pulse generator (LRS), and the POS IN trigger source must be chosen.

In the current test beam this feature is not being used, and the only signal to start readout is the TFIB_trigger. The trigger to start the readout is then generated by a coincidence of scintillators up and downstream.

But, if a charge inject using the pulse generator or a radioactive source wants to be done, the setup must be the following:

- Initialize FPIX0 (injecting the cells you are interested);
- Take the Chip_or signal in the counting room (NIM signal, #13 in the panel), and connect it in the STAR input trigger (#14 in the same panel);
- Connect the pulse generator (with the trigger in LINE mode) in the analog input in the inner board. You can also use the radioactive source, and before start the readout you should see Chip_or going low for 200us and then going high again.
- "Begin run" the in STAR software.

What's wrong if Chip_or doesn't go low or it goes low and high like crazy?

If Chip_or is not going low you have to check the following points:

- If the threshold is too high the chip will never fire. By the other hand, if the threshold is too low, FPIX0 will oscillate. We've been using 2.05V as the lower limit, and 1,5V as the number to use in the initialization process.

- Take a look in the level from the pulse generator. If it's too low the chip will not fire.

- **DON'T INJECT NEGATIVE CHARGES! BEFORE CONNECTING THE PULSER GENERATOR MAKE SURE THAT THE OFFSET IS HIGH ENOUGH TO AVOID A NEGATIVE SIGNAL IN THE ANALOG INPUT OF FPIX0**
- ***Keep in mind that the threshold has an inverted relation: a low threshold means a high voltage (~2V), and vice versa.***

Signals of interest

Several signals in and out the outer board were added in the original design, and they don't appear in any document but here.

Signal	Board	Type	Pin number
STAR trigger output	Outer	TTL output	Altera[43]
STAR accept trigger counter input	Outer	NIM input	LEMO3
Pulse Generator trigger	Outer	TTL output	Altera[41]
TFIB_TRIGGER	Outer	NIM input	LEMO1
"Fast" Chip_or	Inner	TTL output(can not drive a 50Ω load!)	Wire connected near the chip, with the silk screen CHIP_OR

Problems using the CiS ST2 board

Recently I found that the CiS ST2 board has a "minor" problem. For some reason unknown so far (ESD?) the Chip_or in this board is always 0, which means that the chip seems to be always with a hit. The problem is most likely to be located in the digital buffer of the chip, once I already killed all the cells and Chip_or still 0.

Fortunately this is not a killer to this board. I already did some tests (not many) and the readout logic (except Chip_or) is working fine. The bottom line is that we can still use this chip in the beam, but some precautions must be made.

- Chip_or no long can be used as the trigger source for readout.
- The same firmware in the outer board can be used, but it will have a "false event" (0000 0000) when the chip is empty, and will work fine when the chip has data.

Where you can you find parts and information about these boards?

All the schematics and parts for the outer and inner boards are located in my office, in the "help desk". Under the desk you can find a box with all the spares for both boards. On this table you will also find a blue folder, where I keep all the commented schematics of the boards, including the changes that were made in the past.

- How about the firmware?
The firmware, the most important piece of information in the outer board, can be found in the following place:

Eseserver0\ese\projects\btev\engineeringprojects\pixel\pixel(misc)\system\teststand_iii\prb \readout\readout.vhd

The whole thing was written in VHDL, but if don't know it don't worry. If you have an idea of AHDL or C you won't have any problem to understand it. I know that the comments there are far from wonderful, but I'll try to work in this point later this week.

If some change has to be done, don't forget to compile everything before programming the board. If you don't know how to do this please ask for some help, because this explanation can take several pages!

Also in the left side of the monitor in my office there are two black boxes. One of them has the CiS ST2 detector board, and the other one has a good outer board. This should be used as a spare for the one we have now installed in the beam.

Problems found so far

I decided to write some lines to have documented the problems that I found in the pixel setup. Fortunately it will be a short cut for future users.

So far all the problems found in the boards seem to have their source in ESD, which means that EXTREME care must be taken when handling these boards. Ground straps are available in the area, and please take 5 seconds to put it in your wrist.

One problem that was recently found was that some chips in the inner board were burned. The symptom for this problem is that the data from the ADC (pulse height) is always 0. The solution is simple: change the chips. The components involved in this case are the buffer (BUF600), the OPAMP (AD817) and the ADC itself (AD775). These are SMD parts, so one has to be careful with the board when perform the change, to do not damage the pads.

The other problem is that Chip_or is stuck at zero, which was already explained.

Power supplies

An important indication that something can be wrong is the current and voltage in the power supplies. A brief description of what do you should to expect follows:

Name	Board	Voltage [V]	Current
VCC	Outer	5.0	~1.2A
-VCC	Outer	-5.2	~50mA
-DVDD	Inner	-7.0	~15mA
AVCC	Inner	5.0	~60mA
V_{th}	Inner	1.5~2.05	0 in the display
V_{guard}	Inner	0.6	0 in the display
*Detector Bias	Inner	-140V	-25nA
IBP	Inner	~	-15uA
IFB	Inner	~	-6nA

Another very important issue about the power supplies is the power on sequence. If you don't want to have the chip oscillating and spikes in the 3V-power supply, the indicated sequence must be followed:

1st – Digital and analog power supplies (VCC, -VCC, -DVDD, AVCC, V_{th}, V_{guard});
2nd – Current power supplies.

- The high voltage power supply to bias the detector must be turned on in steps of ~-25V. Starting from 0V the user can check if the current is OK. This current is very sensitive to temperature and light variations on the chip. In the test beam, with the boards out side the hut, a current of almost –40nA could be seen.